

Reports File  
No action  
add

**GEORGIA INSTITUTE OF TECHNOLOGY**  
OFFICE OF RESEARCH ADMINISTRATION  
**RESEARCH PROJECT INITIATION**

Date: October 15, 1973

Project Title: "Development of a Computer Technology Lecture/Laboratory Curricula and Materials"

Project No: E-26-509

Principal Investigator Dr. Don Harmer

Sponsor: Digital Equipment Corporation

Agreement Period: From 10/1/73 Until 9/30/74

Type Agreement: Research Agreement and Addendum dated July 9, 1973

Amount: \$20,000

Reports Required: Progress Reports, monthly

Sponsor Contact Person (s):

Russell Cramer

~~Mr. Alan Nowbray~~

Manager, Product and Sales Support

Education Products Group

Digital Equipment Corporation

146 Main Street

Maynard, Massachusetts 01754

Assigned to: Nuclear Engineering

COPIES TO:

Principal Investigator

School Director

Dean of the College

Director, Research Administration

Director, Financial Affairs (2)

Security-Reports-Property Office ☒

Patent Coordinator

Library

Rich Electronic Computer Center

Photographic Laboratory

Project File

Other \_\_\_\_\_

2nd  
M-336

No action  
action  
OK

GEORGIA INSTITUTE OF TECHNOLOGY  
OFFICE OF RESEARCH ADMINISTRATION  
RESEARCH PROJECT TERMINATION

Date: 3 September 1975

Project Title "Development of a Computer Technology Lecture/Laboratory  
Curricula and Materials"

Project No: E-26-509

Principal Investigator: Dr. Don Harmer

Sponsor: Digital Equipment Corporation

Effective Termination Date: 21 August 1975

Clearance of Accounting Charges: 31 January 1975

Grant/Contract Closeout Actions Remaining:

Assigned to School of Nuclear Engineering

COPIES TO:

Principal Investigator

School Director

Dean of the College

Director of Research Administration

Office of Financial Affairs (2)

Security - Reports - Property Office ✓

Patent and Inventions Coordinator

Library, Technical Reports Section

Computer Sciences

Photographic Laboratory

Terminated Project File No. \_\_\_\_\_

Other \_\_\_\_\_

GEORGIA INSTITUTE OF TECHNOLOGY  
ATLANTA, GEORGIA 30332

SCHOOL OF  
NUCLEAR ENGINEERING

July 31, 1974

Mr. Alan Mowbray  
Educational Products Group  
Digital Equipment Corporation  
146 Main Street  
Maynard, Massachusetts 01754

Dear Mr. Mowbray:

Re: "Development of a Computer Technology Lecture Laboratory  
Curricula and Materials" -- Project E-26-509  
Progress Letters #1 and 2 Covering Oct.-Nov. 1973

In reference to our lecture/laboratory curriculum development, it does appear that a simple introduction of the concepts of programming (e.g., resolving a complex operation into a linear sequence of simple operations combined with repetitive cycles of groups of simple steps) would be of benefit to the course. Such an introduction could be taught "a la" programmable desk calculator via FOCAL or BASIC on a 4-K machine. This could provide an introduction to computers at a minimum level and would not necessarily be required if the student has had this earlier (as in many high school programs now).

After careful evaluation of the earlier program, it would appear that a more stepwise laboratory approach that includes more active programming and the driving of actual devices, e.g., lamps, TTY's, etc. would be of value. Particularly if combined with lecture/lab demonstrations would this be useful. To reduce the wiring effort on the part of the student, some prewiring of experiments could be effected to permit the student more time on the essential part of the experiment.

A series of experiments based on practical interfaces could be used for advanced classes or as a quarter or semester project for a basic class. It appears practical to present both for the instructor's choice. Currently we are trying this with selected students.

As was discussed earlier, several items are required for prototype development. Principal among these are the components for the new "I/O Access Box." The basic design of the enclosure was sent under separate cover, and a 1/2 size copy of the drawing is enclosed. It is suggested that this be fabricated as an aluminum sand casting, similar to the original one. Since it can accommodate three H803 connector blocks, instead of two as in the early version, the pricing can be similar.

Mr. Alan Mowbray

July 31, 1974

Progress Letters #1 and 2 Covering Oct.-Nov. 1973

Page 2

The prototype assembly then consists of:

- 1 each Sand Casting as per drawing
- 3 each H803 connector blocks
- 1 segment of "Omnibus" back plane board

(For experimental purposes, it would be useful to obtain two or three of the "Omnibus" back plane boards.

A minor problem exists in usage between signal lines on the "Omnibus" and the external bus, e.g., the "Omnibus" contains 24 signal lines per connector whereas the external positive bus contains 18, the extra lines (6) being ground in the external bus. This in itself presents no particular problem, as these additional grounds may be supplied via the standard Ext Bus IO cable, as long as pin T2 is not grounded in the Omnibus segment. The non-use of pins U1 and V1 likewise produces no particular problem. (These are normally not connected in the Ext Bus to prevent damage from -15V if the cable is plugged in upside-down.) The only apparent problem may be in the use of a signal on B1 in the Ext Bus, whereas in the "Omnibus" this is reserved for a test point in certain modules. Some conflicts could arise from this dual usage. I suggest that this conflict could be resolved by using jumpers for Ext Bus Usages to bus the "B1's" together and to remove them for "Omnibus" usage, if required.

To prototype the "patch panel" quad logic board, the components required are:

- 1 each W968 Collage Board
- 1 box\* panel light socket ferrules (DEC #90-07812, used in front panels, etc.)
- 72 IC Sockets (Solder type) 16 pin

\* Or some large number of ferrules

Other components for this are obtainable locally.

In order to prototype and test the student experiments (particularly student tested), a number of wire wrap quad cards will be required.

If possible, it would be advantageous to have enough boards for a full class, e.g. 15-20 students. W942 boards and/or W952 boards (allow 24 pin IC's) would suffice. I would also like to try several W967's (Omnibus) and/or the later versions which have the bus receiver-drivers on them, M1709, for Omnibus interfacing. The latter may be the most useful way to present Omnibus interfacing. Several W943's would be useful for the special projects, e.g., those that use the last two slots on the system's module.



Mr. Alan Mowbray  
July 31, 1974  
Progress Letters #1 and 2 Covering Oct.-Nov. 1973  
Page 3

Currently there is a need for some H-915 patch cords, 4 inch and 8 inch lengths (these come in packages of 100).

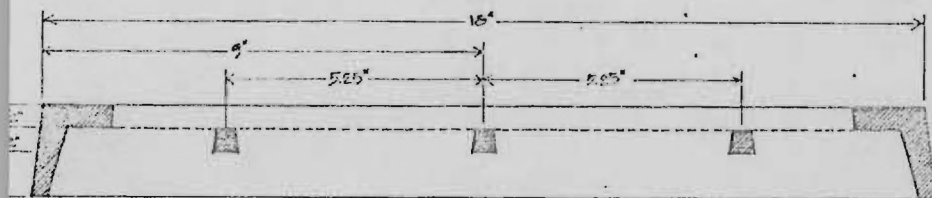
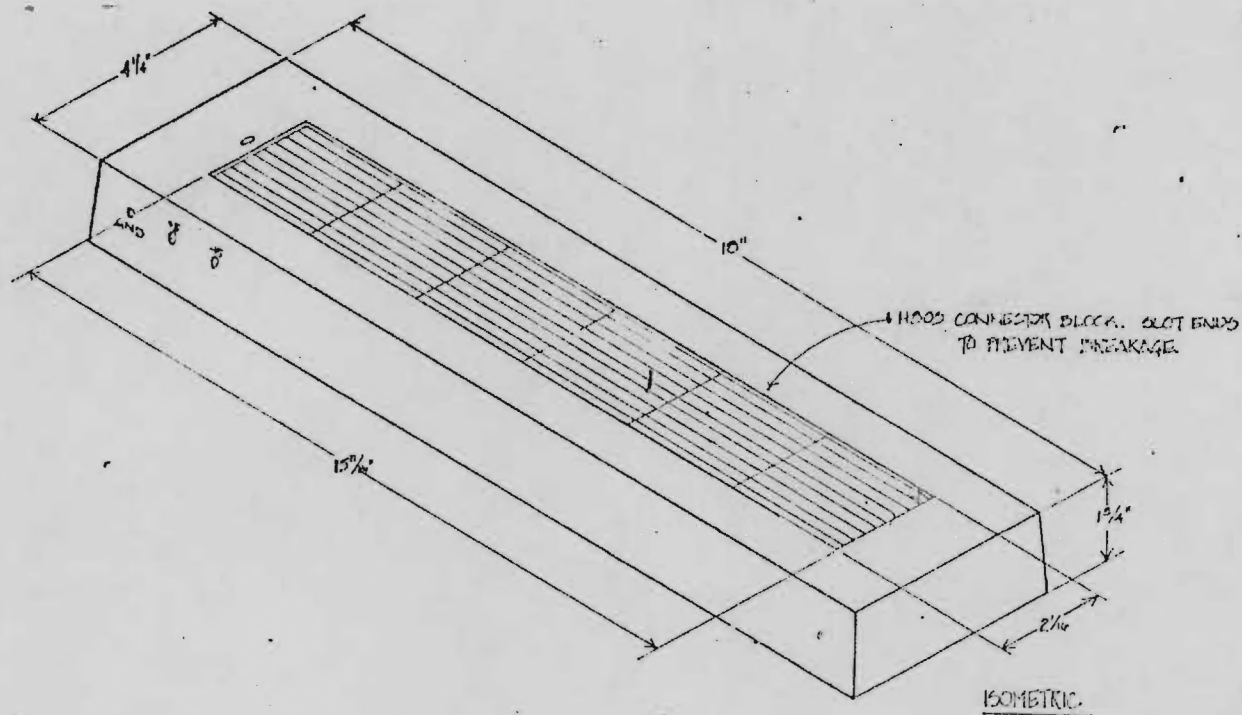
I look forward to your visit to Atlanta when we can discuss the project in detail.

Sincerely,

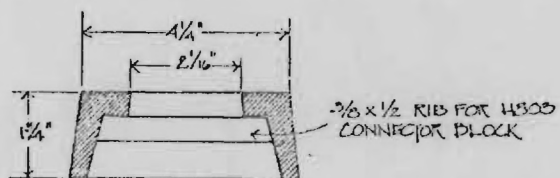
Don S. Harmer  
Professor of Physics and  
Nuclear Engineering

DSH:lsg  
Enclosure

This is a revision of our letter  
dated November 23, 1973.



LONGITUDINAL SECTION



TRANSVERSE SECTION

EDUTECH I/O ACCESS BOX  
TYPE 2

SCALE: 1/2" = 1"

DESIGN: D.S. HARMER

DRAWN: E.M. KILLELEA

DATE: NOVEMBER 11, 1973

DEPARTMENT OF NUCLEAR ENGINEERING  
GEORGIA INSTITUTE OF TECHNOLOGY  
ATLANTA, GEORGIA.

GEORGIA INSTITUTE OF TECHNOLOGY  
ATLANTA, GEORGIA 30332

SCHOOL OF  
NUCLEAR ENGINEERING

August 12, 1974

Mr. Alan Mowbray  
Educational Products Group  
Digital Equipment Corporation  
146 Main Street  
Maynard, Massachusetts 01754

Dear Mr. Mowbray:

Re: "Development of a Computer Technology Lecture Laboratory  
Curricula and Materials" -- Project E-26-509  
Progress Letters #3 and 4 Covering Dec. 1973 - Jan. 1974

As you know our original plan for the project was to develop the preliminary revised experiments during last summer and to debug and test these on a student population during the Fall quarter. As before the plan was to have the students submit their experimental write-ups with a section on "What I wish I had been told before starting this experiment". In this way the student can provide valuable feedback to point out errors, ambiguities and conceptual problems, in addition to the evaluation of the worth of each experiment.

Since the contract was not implemented until the middle of October, and the materials (I/O Box, board, etc.) were not available to my class, we could not do this.

However, we did implement some of the changes we discussed, using the old dual boards. Particularly, we assigned each student one of the possible variations of the experiment, such as using binary to octal decoders in device selection, providing more obvious output devices, etc. In addition, we designed some new more advanced experiments.

Of particular interest was a "time-of-day" clock which could either display (7-segment) or could be read as a teletype like device, that is as ASCII characters on a parallel, one-at-a-time basis. This device could be useful in any system that keeps time-of-day and date, (e.g. OS-8).

In general, the experience with the students this quarter has reinforced my belief that the students show much more interest in practical interfaces than in general purpose interfaces.

As we discussed earlier, the add-on of a Fetch-Stop, Execute-Stop with auto-restart feature would provide the instructor with a valuable teaching/demonstration aid, particularly when used in conjunction with the demonstration register lamp panel. At your request I have kept close touch with your

Mr. Alan Mowbray

August 12, 1974

Progress Letters #3 and 4 Covering Dec. 1973 - Jan. 1974

Page 2

engineer on the Project. The student will also obtain great benefit from using this "programmer's aid" during debugging of his interface. One should consider this gadget as a hardware break-point, a la ODT. As I suggested during my initial proposition to develop such a panel, I strongly feel that the device would have a much larger market outside of the particular application in the EDUTECH package. I believe that in its final form it should be separable from the display panel and its interface so that people doing program development can purchase it at a reasonable price.

I hope that just the switch panel alone could also be purchased so it may be also adopted to 8-I's and 8L's many of which are being used to teach the current version of the course.

The I/O Access Panel could usefully retain the six cable sockets as in the H309 B and A versions. The H309 C version has only 5 sockets in a 5 1/4 inch panel. If the "B" panel were used, relabeled as in the "C" version, a cable from the "Berg" connector on an internal interface could be brought out to the front panel in addition. I think this could be very useful, and produced at only the cost of re-doing the silk screen for the "B" version. Since almost all in the Omnibus interfaces communicate with the outside world via the Berg connector, this would permit some easy student experiments with the I/O signals to and from these devices on a table-top basis.

We had also discussed the role of the "LOGIC LAB", H500, in such a course. I believe it can play a role, possibly even in a version that plugs into the I/O access box for power and connections. Certainly, it can be very useful for demonstrations. However because of its limited complement of IC's its role is limited. As we discussed with Bob Puilliot, I would like to suggest the following add on modifications (see photos attached)

1. provide several patchable IC sockets (six?)
2. provide at least two or more patchable cable sockets  
(so that cables or M and K series modules can be plugged in)
3. provide some D-type flip-flop in addition to the J-K's supplied.

I would be glad to discuss this further with you and Bob's group at the meeting we discussed. Please let me know when you and Bob can arrange this meeting.

We are currently also working on the problem of teaching assembler in an effective fashion and will let you know how it turns out.



Mr. Alan Mowbray

August 12, 1974

Progress Letters #3 and 4 Covering Dec. 1973 - Jan. 1974

Page 3

We still have not received the parts for the new experiments from you. Most of the IC's I ordered directly are here. These are in addition to the complement quoted for the old style kits. Please see if you can expedite the delivery so that we may have them for the Spring Quarter class. Please pursue the ferrules for the patch panel, I am interested in trying this approach out for the early simple experiments with logic and for demonstrations. (See proposal and letter of November 23, 1973.)

Sincerely,

Don S. Harmer  
Professor of Physics and  
Nuclear Engineering

DSH/f

This report is a recompilation  
from notes, telephone logs,  
and informal letters.

GEORGIA INSTITUTE OF TECHNOLOGY  
ATLANTA, GEORGIA 30332

SCHOOL OF  
NUCLEAR ENGINEERING

August 12, 1974

Mr. Alan Mowbray  
Educational Products Group  
Digital Equipment Corporation  
146 Main Street  
Maynard, Massachusetts 01754

Dear Mr. Mowbray:

Re: "Development of a Computer Technology Lecture Laboratory  
Curricula and Materials" -- Project E-26-509  
Progress Letters #5 and 6 Covering Feb.-Mar. 1974

As you know we have been struggling with an approach to teaching assembler programming that has the same concepts as we are using in interfacing. That is, that the student can test himself and receive the reward of success, and that the turn-around-time of this loop should be short. In fact, the rewards should be commensurate with the effort the student puts forth, as should the actual amount learned.

One of the major problems in teaching programming has always been that the student has to write so much before he can test it that he often has forgotten why he did things the way he did. With no boundaries, he does not learn good techniques. Most of the preliminary efforts usually are to write a TTY handler, a console light shuffler, etc. None of these "turn the student on"! The approach which we are trying in the current quarter just ending is described in the attached write-up. As such it represents a good second course for the EduTech System.

The add-ons required for successful use of this approach would be a TD8e dual drive, an additional 4K of memory (e.g., 8K), EAE and TD8e ROM; that is, a minimum OS/8 system, with EAE. A scope with appropriate D to A interface is required, but this could be one of the student projects in the interfacing course. The scope required for debugging would be available for display use in this course until the group can afford a VR-14 or equivalent.

This software approach also suggests several interfacing experiments which also have practical value for the instructor:

1. 8 to 10 bit D to A (dual) for a scope display interface (using packaged D to A's such as Analog Devices, Analogic, etc.)
2. 6-bit A to D for joysticks
3. Digital input for sense switches.

Mr. Alan Mowbray

August 12, 1974

Progress Letters #5 and 6 Covering Feb.-Mar. 1974

Page 2

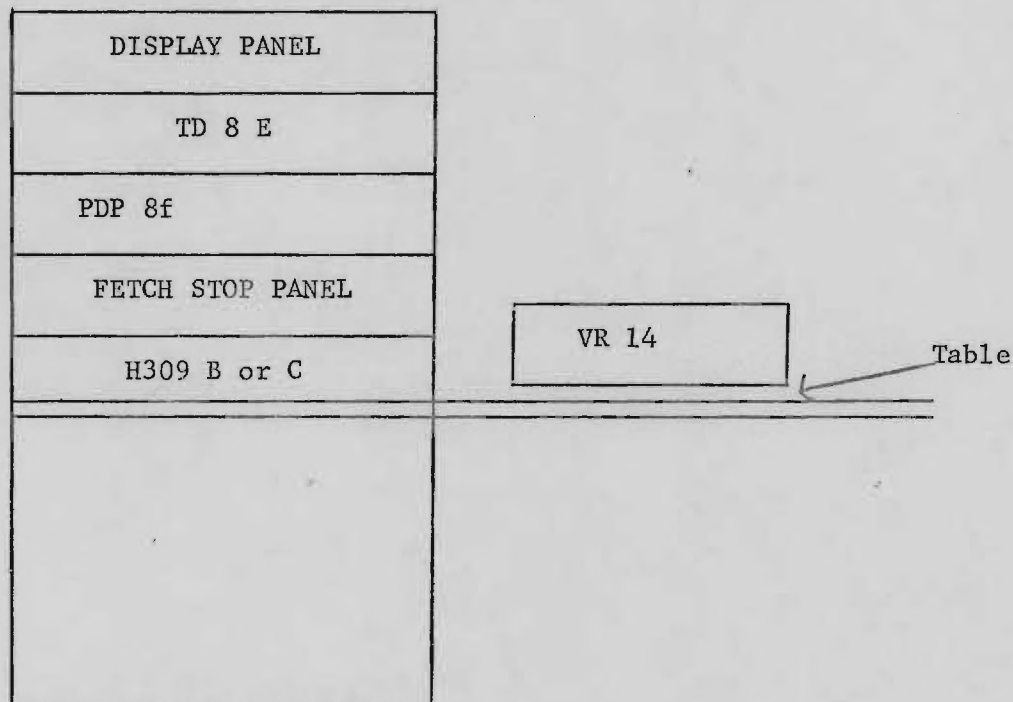
Each of these has tutorial value in connection with the material to be covered in the text and has practical value in later use in the course. The instructor can retain for later use the "best" board out of a student group. (Often a further incentive to excel by the student.) The further advantage is that the student, on completing his interface, can see its effect on a visually oriented, running program.

#### POSSIBLE EDUTECH PACKAGES

1. Bare minimum system -- for interfacing only (AC transfer only)
  - PDP 8/f without TTY (4K)
  - External I/OBUS with
  - H310 B MODULE TEST ASSEMBLY
  - (new hex I/O Access Box)
  - 10 to 20 H311 Logic Design Laboratory kits with W9\_\_\_\_\_ quad boards
  - instead of W979 dual boards
  - Option add TTY
2. Above plus: display panel
  - H309 B or C I/O access panel with 5 volt power supply
  - (H 710 or equivalent)
  - Programmer's Aid Panel
  - (Fetch-Execute Stop)
  - tall cabinet -- (TABLE ?)
3. Same as 2 but with 8K PDP 8/f
4. Same as 3 but including TD8e and TD8e ROM and OS-8
5. Add EAE, (SPACE-WAR Programming system)
6. Add VR 14 - VC8e
7. Standard EDU Software packages, e.g. multi-user Basic etc.
8. Add RK-8
- etc.

For interfacing convenience the location of components within the cabinet should be:

Mr. Alan Mowbray  
August 12, 1974  
Progress Letters #5 and 6 Covering Feb.-Mar. 1974  
Page 3



Note that the above at level 4-5 can also now be used as a full programming machine for teaching of assembler and high level languages.

I am aware of software developments in the works which could possibly provide a basis for an extensive multi-user system also.

The packages suggested are to permit low-cost entry with upgrades on a yearly basis, as is necessary in college, university, tech school environments.

We still have not received any of the materials you were sending for the prototype experiment development.

Sincerely,

Don S. Harmer  
Professor of Physics and  
Engineering

DSH:lsg  
Enclosure

This report is a recompilation from notes, telephone logs, and informal letters.



## Teaching Assembler Programming via SPACEWAR

D. E. Wrege and D. S. Harmer  
School of Nuclear Engineering  
Georgia Institute of Technology

A novel approach to the teaching of computer programming at the assembler level has been tried with great success at Georgia Tech. Faced with the problem of covering in a ten-week quarter, assembler level programming, real-time programming, and operating systems, a method of approach was developed based on modular programming concepts and utilizing a maximized "hands-on" approach. The need for this type of course developed out of a minor option in "Computer Engineering" for all engineering majors.

Small computers (often called "mini-computers") are being found in ever increasing numbers of applications, especially in technological areas. The motivating factor behind this development is two-fold. First, general purpose minicomputers, as a result of mass production techniques and the technological advances in electronics are less expensive and more reliable than the "hard-wired" equipment that they are replacing. Second, there is a degree of flexibility to be gained from computers that is not available in conventional approaches. This flexibility is increasingly desired in a rapidly changing technical world. As a result of this trend it is becoming increasingly important that students in technological disciplines understand how to use small computers for their applications.

The depth of understanding required by these disciplines is generally not that of a hardware designer or software systems engineer, but rather enough knowledge to enable them to communicate with specialists in these areas.

Enough familiarity is required to realize what can be accomplished and how to specify meaningful algorithms. As an attempt to satisfy this need a computer engineering minor is being developed at Georgia Tech with a "hands-on" approach, introducing the students to both hardware and software concepts. The program is aimed at the graduate level engineering student with little or no prior training in digital design or computer science.

The first course in the series is aimed at an in-depth understanding of hardware and interfacing and the hardware-software "trade-offs" (Experiments in Logic Design and Computer Interfacing, D. S. Harmer et al, Digital Equipment Corp., Maynard, Mass., 1971).

The second course in this series is in assembly level programming techniques oriented primarily toward "on-line" or "real-time" applications. With the current trend in small computer software the use of machine language programming as an approach to solving problems is de-emphasized. High-level languages with real-time capabilities have been developed for many mini-computers, and the trend is to use these languages for real-time applications. What is often overlooked is the fact that some assembly-level (machine-language) programming needs to be done to tailor the high level language to the real time application undertaken. Once this is accomplished the majority of the remainder of the programming task may be accomplished in the high-level language itself. Unfortunately, although the machine-language portion of the coding effort may be only 5-10% of the programming effort, a comprehensive understanding of this subject is required to accomplish the desired end. Not to mention the fact that to understand how to modify the high-level language often takes a detailed understanding of assembly level programming.

It was with the above motivation that the introductory course in software

systems is oriented toward machine-language coding techniques. The mini-computer used was a PDP-8 family machine manufactured by Digital Equipment Corporation. This computer is especially well suited for this type of instruction as it has a small instruction set and simple addressing schemes. Similarly the I/O structure is uncomplicated and easy to understand.

The first attempt at teaching this material proved to be less productive than was anticipated. After a great deal of retrospective analysis as to what caused this failure, the approach toward teaching this subject was completely changed and the second time the course was taught we experienced a vast improvement in response. The main purpose of this report is to relate our findings and discuss this new approach.

We found that although theoretical discussions were asthetically pleasing the student had little to relate this to and thus the educational value was minimal. This lead us to appreciate the value of a hands-on approach to learning, placing an emphasis on experience. Actually, the fact that programming techniques are best learned by writing programs is not new, and we apprecaited this fact before we attempted the pilot run of the course. What was not realized was how important the program content was. For example, a student will be better equipped to undertake real-time programming tasks with 50 hours of time spent writting non-real time oriented programs than one with 25 hours experience doing real-time oriented exercises. What seems to be important is the concept of how to break problems down into a form that lends itself to machine-language coding techniques. Therefore the most important single factor in the design of this type of instruction is to motivate the student to spend as much time as possible writing code himself.

There is a major problem in the development of exercises for a course

spanning a 10 week period. Namely that assignments must be doable over an approximate 1-2 week span. The result is that programs tend to be short, and unsophisticated. For example, an integer-octal-calculator with add, subtract, and multiply is the most difficult assignment possible. This leads to a problem regarding motivation. When a student completes a task such as this, he is not particularly overwhelmed with his result. Assignments become homework instead of fun.

In order to motivate the student, it was decided that the assignments should involve writing a game that was sufficiently cute as to make development of the program fun. This program should be modular to the extent that every assignment ended with a working program with direct feedback to the student as to how he was performing. Also there should be some degree of individuality available so that the student could exercise creativity as he developed sequential modules of code. Finally, if possible, the program should involve many examples of the type of programming techniques that wish to be taught.

To accomplish this end a spacewar program was written, in a highly modular fashion, that had the above characteristics. Spacewar has two spaceships operating under the influence of gravity from a star. The ships obey all of Newton's laws (approximately) and fire missiles which also move under a gravitational field. Many options (implemented under interrupt control) are available to demonstrate the use of interrupt service routines and I/O overlap. These options include changing the strength of gravity, the types of ships displayed, thrust constant rockets, conservation of rotational momentum of the ships, and reflection at the scope screen edge. An added advantage of these options is that they demonstrate to the student that he may be



creative in writing his own version of the game.

In order to have positive feedback to the students, with a working program on completion of each assignment, subroutine modules are given out with each programming task. We have found that the student learns a great deal from studying these subroutines, in addition to what they learn from experience. For example, one of the first assignments is to display a space-ship using a supplied display subroutine. Next they make it rotate, and are given the routine to display a ship. Then they are given a rotation subroutine when they are asked to make it translate. Thus at every stage one of the routines that they are supplied with is a subroutine that performs what they have just accomplished. We have found that this approach is extremely useful since the student usually desires to use what he has just finished and will therefore study the supplied routine to see if it is really better. There is no doubt that one of the best ways to learn computer programming techniques is by trying to understand how someone else programs. A second feature of these techniques is that the students are continuously trying to outdo one another. Consequently, they spend more time learning than they realize. For example, it was common for a single student to write programs to display three or four different ships in an attempt to outdo his friend. The visual feedback of the display formed an invaluable part of the learning experience.

The result of our experience has been that it is almost impossible to "teach" a student machine language programming techniques. What needs to be done is that the instructor should motivate the student to teach himself, supplying guidance and help along the way. We have found that approximately one-half of the contact classroom hours are spent answering questions, which

means that the students are interested.

This approach has proved to be highly successful for an introductory course in assembly level programming, bringing the student rapidly to the point where more classical teaching techniques may be used.

GEORGIA INSTITUTE OF TECHNOLOGY  
ATLANTA, GEORGIA 30332

SCHOOL OF  
NUCLEAR ENGINEERING

August 20, 1974

Mr. Alan Mowbray  
Educational Products Group  
Digital Equipment Corporation  
146 Main Street  
Maynard, Massachusetts 01754

Dear Mr. Mowbray:

Re: "Development of a Computer Technology Lecture Laboratory  
Curricula and Materials" -- Project E-26-509  
Progress Letter #7 Covering April 1974

Enclosed is the abstract of the paper on the course development to be presented at DECUS. I think it would be advantageous to have the prototype system available for demonstration in the show. I am sure it will elicit significant interest. Could you provide me with slides of the prototype display panel, fetch-execute stop display panel, and H309C for use in this talk? I think you will find enormous interest from those in unrelated areas in the programmers' aid panel.

It appears that the legal problems associated with the no-cost lease have been resolved and the contract will be signed soon. However, it turns out the Georgia law requires that DEC sign the lease before it can be executed here. We are awaiting final approval from the EDP committee of the State, but they cannot consider it until you sign it. The contract has been returned to you via air mail, special delivery for execution. Please return it to my home address to expedite its return. It must reach us in time for the next EDP committee meeting, otherwise it will be further delayed.

COURSE DEVELOPMENT

The major goal of the course is to provide a deep association and understanding of sequential combinatory logic, from the hardware to the software, including the hardware/software trade-offs. Throughout the course, the aim is to provide the association between what is happening in the computer (instructions being executed, etc.) with what is happening in the logic, e.g., interface, I/O Bus etc. In every case, it is emphasized that linear combinations of logic decisions, sequential in time, perform complex operations, even up to the high level language. Major emphasis is placed on viewing the computer operations from Gordon Bell's register-transfer description, and in describing computer instructions in terms of the Source-Operation-Destination view.

The curriculum approach is based on demonstration/discussion of basic principles followed by the student verifying and applying these principles in the laboratory. The experiments and demonstrations are to be designed so that the student can verify the principles learned and apply them in small increments, and test his knowledge as he goes. A short turn-around feedback loop to success is essential to his building cumulatively his understanding. To this end the experiments are designed to provide a cohesive stepwise approach to developing complex interfaces. For example, in discussing AND gates, the student will build the first elements of a device decoder, and the lecture will point out and discuss that instructions are decoded by the processor in a similar manner. As the student progresses through the course he will add to his interface board, building from device decoding, to flags, to registers, to interrupts, etc. In parallel, the software required to make the logic perform will be discussed, and the student will incrementally learn the application of the instructions and the basics of machine language instructions. Each experiment will introduce some new instructions (e.g.: CLA, TAD, IAC) to be applied to the particular equipment.

The outline of the course is similar to the earlier version, with extensive revisions and additions to the demonstrations and experiments.

- A. Initial Demonstration of Computer: BASIC - FOCAL - MUSIC (SPACEWAR), GAMES, etc.  
Goal: Familiarize the student with the processor and arouse the students' interest in learning to use it.
- B. Familiarization:  
Use of switches, scope, observe I/O pulses, etc.
- C. Instructor's program to see instructions being executed under various conditions, IOT's and timing pulses
- D. On-line: demonstration of Binary - Octal numbers and conversions; in lab students enter various numbers on switches and are prompted by processor
- E. Demonstration of Instructions in Single Step: e.g.: contents of registers at end of each cycle
- F. Instruction and Device Decoding - AND gates - selection of unique bit pattern - binary to octal decoding with AND gate logic - the IOT instruction
- G. NAND gates
  - 1. inverter concepts - NOT
  - 2. inverters from NAND's
  - 3. Anding with IOP's for time synchronization
  - 4. The AND function as coincidence device for performing time sequential operations
- H. The AND as a logic test
  - 1. Test the condition of a signal
  - 2. The SKIP function as a logical branch in software - e.g.: SKIPIFFLAG
  - 3. The AND instruction



- I. OR gates (and NOR's)
  - 1. clear flags on IOT2 or 6
  - 2. clear AC on IOT2 or 6
  - 3. LOAD signal on IOT4 or 6
- J. Flip-flops and Registers - memory
  - 1. Provide holding register to retain transient information (e.g.: the condition of AC bit 11)
  - 2. Develop and introduce the TAD instruction, CLA, IAC (NOTE: this 1-bit register is used later as INTERRUPT ENABLE)
  - 3. The 1-bit register is set or cleared according to condition of the switch register bit 11.  
The IAS instruction: use single step to show that the condition of the register is not equal to the switch, but that several program steps must be executed to transfer from switch register to device register.
  - 4. Use switches to set timing loop via ISZ to show bit is really remembered by device. Introduce ISZ, CMA.
- K. Introduce multi-bit registers (e.g.: keyboard, paper tape reader)
  - 1. Add 8-bit input register to interface - demonstrate input transfer via program control
  - 2. Binary bit patterns can be defined as characters (e.g.: ASCII or other coding)
  - 3. Demonstrate that under these circumstances the computer has no way of determining that an external event has occurred except by an explicit software test
- L. Interrupt:
  - 1. Wire Interrupt and Interrupt Enable
  - 2. Demonstrate what happens if INT REQ is grounded (use single step and JMP - 1 or similar program)  
Show location 0=PC after interrupt and that PC=1
  - 3. Student does input from keyboard under Interrupt control.
- M. Interrupt Devices
  - 1. Student builds simple clock for interrupts
  - 2. SCALER or Event Counter (count to overflow and then interrupt)
- N. Output Devices (AC Transfer)
  - 1. Student adds circuitry to load counter above from AC
  - 2. Tests by loading compliment of desired count and counting to overflow
  - 3. Present the concept of negative numbers as compliments
- O. SHIFT REGISTERS
  - 1. Use clock for timing of shifts
  - 2. Use of shift register for parallel to serial conversion
  - 3. Output characters to TTY
- P. Memory and Memory Organization
  - 1. Memory addressing schemes
  - 2. Paging and Page relocation  
(Test Program written for page 0, then page 1, then page X)
  - 3. Possible experiment - ROM or PROM blaster

Mr. Alan Mowbray  
August 20, 1974  
Progress Letter #7  
Page 4

- Q. A to D and D to A conversion
  - 1. SCOPE Display interface
  - 2. Joy stick interface
- R. Data break Devices  
(Under revision and development)

Several advanced experiments are under consideration and development, including a time-of-day clock and alternate versions of the experiments to illustrate the major similarities of interfaces (e.g.: keyboard, paper tape Reader and Digital Input; Digital Output, D to A register, and TTY output register, etc.).

The above will provide an idea of the approach and some of the experiments. It is planned to refine these and provide a self-consistent combination of demonstrations; hardware construction and testing and software introduction, with each step providing the student with active feedback.

I look forward to discussing the progress with you prior to the DECUS meeting at Maynard. I would hope the sand castings of the new I/O Box will be available then so I can finish the development of this vital accessory to the course.

Sincerely,

Don S. Harmer  
Professor of Physics and Nuclear  
Engineering

DSH:lsg  
Enclosure

This report is a recompilation from  
notes, telephone logs, and informal  
letters.

# Computer Technology Lecture/Laboratory

## Curricula and Materials

Don S. Harmer

School of Nuclear Engineering

Georgia Institute of Technology

Atlanta, Georgia

and

Alan Mowbray

Digital Equipment Corporation

Maynard, Mass.

The teaching of a broad spectrum of topics in computer technology and applications appears to be most effective when coupled with extensive "hands-on" experience by the student. It is particularly effective when the student's effort is commensurate with the amount of learning and when the reward of success is given for each increment of learning. Integration of lecture and laboratory material is important to prevent the student from wandering blindly through the lab work, "Betty Crocker" style, because he does not have the information at hand to understand and absorb the principles being illustrated. A major factor in learning has been found to be a feedback mechanism which points out to the student quickly as he learns each new concept where the misunderstandings and misconceptions are in his new knowledge.

The course, which was developed as an in-depth introduction to computer technology and applications, attempts to use the above principals in its philosophy of presentation. The laboratory manual for this course "Experiments in Logic Design and Computer Interfacing" has been used extensively at Tech and many other Institutions. A major revision of this course material is underway. The materials will be considerably revised to take advantage of the new technology. The laboratory experiments will be based on a new I/O Box with hex-sockets, permitting External 8-bus, "Omnibus" <sup>TM</sup>, and "Unibus" <sup>TM</sup> interfacing. The laboratory experiments will be extensively revised to include a wider variety of basic logic experiments and experiments utilizing external bus and "omnibus". A major improvement in the textual part of the manual will be made to include a much more extensive section on digital logic and more on computer architecture, advanced interfacing concepts and software.

The structure of the text and experiments will be modular so that it may accommodate a wide variety of applications, in Vocational-Technical Schools, computer Science programs and as an introduction in computer Engineering Applications.

GEORGIA INSTITUTE OF TECHNOLOGY  
ATLANTA, GEORGIA 30332

SCHOOL OF  
NUCLEAR ENGINEERING

August 21, 1974

Mr. Alan Mowbray  
Educational Products Group  
Digital Equipment Corporation  
146 Main Street  
Maynard, Massachusetts 01754

Dear Mr. Mowbray:

Re: "Development of a Computer Technology Lecture Laboratory  
Curricula and Materials" -- Project E-26-509  
Progress Letter #8 Covering May 1974

The wire wrap boards finally arrived the second week in May. There is enough time for the students to try out at least one of the experiments on the quad boards. Half of the students will try out a keyboard input experiment using one of the low cost keyboards available from surplus houses (~ \$35) and the other half will interface a paper-tape read head. The initial interface will be to the external bus. These experiments are now in progress.

The basic experiments are being revised to fit the new format. After much consideration, the new experiments will be presented in both omnibus and external bus form. This is being done for several reasons:

1. To show the commonality of the basic interface logic
2. To provide an introduction to Omnibus (and Unibus) bi-directional bus concepts
3. To provide in one place a commentary on the similarities and differences of the two schemes
4. To provide a basis for discussion of bus loading and termination.

The logic diagrams for the experiments are being laid out so that either Omnibus or external bus can be wired and the common logic elements are in the same locations on the board. This will facilitate discussion and understanding of the differences and the similarities between the two systems. I think the approach of showing both schemes will be of long term benefit, since the student will be able to examine both in detail without having necessarily to wire both. In general, he will wire to the external bus, since this requires less hand wire-wrapping and provides a "safety-factor" against damage to the processor by serious wiring errors. Only after he has acquired reasonable skills will he wire to the omnibus itself. However, the presentation will be such that, if the instructor desires, the student may wire directly to the omnibus from the beginning. The design of the I/O Access box will permit this



Mr. Alan Mowbray  
August 21, 1974  
Progress Letter #8  
Page 2

(and also to the Unibus if desired with some additional logic). The enclosed rough diagram of the keyboard experiment will give an idea of the logical progression of the development of a full interface in a stepwise manner as outlined in Letter #7.

The student first wires the multi-input NAND(A) as an example of the application of AND'S (e.g., device decoding), then as illustration of binary-to-octal and instruction decoding (B), inverters and NOR'S (C), time sequencing and coincidence (D), FLAGS and SKIP(E), registers (G) and (F), interrupts (G). Note the easy progression in sequence, with testing and debugging at each step. I also plan to provide wire-lists to aid the instructor in debugging troublesome student problems and to assist in providing a uniform layout of the student experiments.

The presentation on the course and the new hardware at the Fall DECUS Meeting was received well, with surprising attendance for a paper being presented on the last afternoon of the meeting. There was considerable interest from those outside of the teaching area in the hardware. Those who do interface development were particularly interested in the I/O Access box which could be used for both busses (and for Unibus); this would aid in prototype interface development. Suitable pricing and marketing would see a large use of this aid.

As I predicted, there was an enormous interest in the programmers' aid panel, particularly among those in industry who do program development for small real time systems. The idea of a hardware ODT, using no core, was received enthusiastically. Of course, those who had used a LINC-8 or 12 in the past were the most enthusiastic, with several in the audience wishing to order one for their 8's immediately. Considerable interest was expressed in the proposed "patch-panel" quad board which would use 22 or 24 gauge wire for patch-cords. This board is yet to be developed; it is awaiting the ferrules (lamp socket eyelets) which you were going to supply (see November letter for DEC part No). This board could be used to particular advantage in the early part of the course and for preliminary experiments. Over all, the presentation was successful, particularly with having the prototypes Display Panel and Programmers' Aid Panel to demonstrate.

As we discussed, I would like to obtain one of the switch panels (without interface) in order to develop a similar interface for the 8/I. Please note that it is impossible to buy such things directly since those parts are not catalog items. As was planned in setting up the contracts, those parts, for prototype development would be obtained directly, rather than through the small parts budget of the contract, since the cost would be considerably lower. I would not be surprised, however, if there would not be a market for the switch panel by itself for an auxiliary switch panel (e.g., sense switches, etc.) since it so nicely matches the decor and color scheme of the 8e,f's.

Mr. Alan Mowbray  
August 21, 1974  
Progress Letter #8  
Page 3

As you requested, I will follow up on the use of Southern Tech as a test site for the curriculum. This is of particular advantage, since they are a division of Tech and I am well acquainted with some of the people who would be interested. I will see if John Keown can attend the summer course for VO-TECH instructors. As I mentioned, it appears that they have some money to contribute, which should make it a lot easier to use them as a test site, particularly since they can cover the cost of the processor and some add-ons. I would hope that they could get at least a TD8E and bootstrap, so they could run OS-8 and teach the programming course in addition.

I will be at CCUC/5 at the end of June and will discuss some aspects of the course in my paper there. I would like to suggest that we plan to have an EDUTECH System at the Fall DECUS, and if so, we can provide some good demonstration software.

Sincerely,

Don S. Harmer  
Professor of Physics and Nuclear  
Engineering

DSH:lsg  
Enclosure

This report is a recompilation from  
notes, telephone logs, and informal  
letters.

GEORGIA INSTITUTE OF TECHNOLOGY  
ATLANTA, GEORGIA 30332

SCHOOL OF  
NUCLEAR ENGINEERING

August 21, 1974

Mr. Alan Mowbray  
Educational Products Group  
Digital Equipment Corporation  
146 Main Street  
Maynard, Massachusetts 01754

Dear Mr. Mowbray:

Re: "Development of a Computer Technology Lecture Laboratory  
Curricula and Materials" -- Project E-26-509  
Progress Letter #9 Covering June 1974

After reviewing the experience with the quad boards during the last three weeks of the quarter, I wish to make the following recommendations for the new student kit design (i.e., new version of the H311 Logic Design Laboratory kit).

- a. The kit box is satisfactory and will hold a quad module
- b. Use a W943 Extended length quad height wire-wrap board (no sockets)
- c. Supply about twenty IC solder type sockets (unsoldered). Use Texas Instruments or AMP low-profile type which have contact springs on both sides of pin. Do Not use type with spring contact only on one side of pin as on the current W942 boards. (These are a disaster in student hands, it is extremely difficult to insert IC's in these sockets without damaging the pins. Our experience in just three weeks indicated that 90% of the student circuit problems arose from these sockets vs 1-2% from the AMP type.) (DEC Part #954 DIP SOCKETS)
- d. In the vial, put 5-10 wire-wrap pins and four H915 8-inch patch cords. The students do break and pull out of the board the wire-wrap pins. Some spares are needed.
- e. The cut-out for the wire-wrap tool needs to be revised to fit the new style tool DEC is currently supplying (e.g., the H811A wire wrap tool). The cut-out for the H812A unwrap tool and the wire stripper are satisfactory.

The kits obviously need to be repriced since the more expensive wire wrap sockets and the large number of loose wire wrap pins have been deleted, and the W943 board is being substituted for the W979 collage board.

As was discussed during my visit in May, we are awaiting the arrival of the sand castings so we can finish the development of the I/O Access Box. (Module Test Assembly, H310 devised to Hex-wide configuration)

I have obtained some boards that could be used to parallel bus the I/O Access box. I am still interested in using segments of the omnibus backplane, as it would not require any new layout. As was discussed earlier, the I/O



Mr. Alan Mowbray  
Progress Letter #9  
August 21, 1974  
Page 2

Access box needs to be bussed so that the wire-wrap pins are still accessible for additional wiring to the end of the block (the dual-width, unbussed segment). Would you try again to obtain a segment of the backplane.

The H309A,B,C I/O ACCESS PANELs as are now being shipped appear to have two problems. First, the sockets on the front panel should be slotted to prevent breakage as cables are removed. This is essential or shortly at least one of the sockets will break allowing some side 1 and 2 pins to contact each other, producing strange results in the processor when the cables are absent. Note that this slotting is done in many other cases (extender boards, etc.). Second, the H309's are being shipped with 24 Volt bulbs (GE-CM327) instead of about 10 volt bulbs. If the pilot lamp were taken off the output side of the regulator (+5V) instead of the input side (~8V) then 6 volt-low current lamps could be used. As it is, one cannot see the pilot lamp and there is always confusion on whether the power is on or off, a great frustration to students. The H309 problems need to be relayed to the appropriate product line for correction.

After seeing the 8A, I would like to suggest its inclusion in the course and EDUTech package. The interfacing can be done and the programs written to test it with as little as one page (two pages would be better) of memory. I think this would be better than the microprocessor approach suggested earlier by Logic Products. Use of minimum 8a would allow low cost expansion to several processors for large classes. One would like to have an 8f (8K or more) system with DECTAPE (with Programmer's and Display Panel) for programming and demonstration (in addition to use in the course). Then one could expand with 8A's with minimum read-write memory. One wonders if there is sufficient power in an 8f or more likely an 8e to drive an 8A. If so then one could expand by adding 8A's without power supply. Perhaps a suitable binary switch and light panel could be devised. Note that the 8A could plug directly into the I/O ACCESS box and with suitable power, run there, with room for student interfaces. Note that this would be excellent for teaching processor maintenance. In any case, I think the role of the 8A warrants further consideration.

I am still awaiting the sand casting to finish the I/O Access box development. Since a large number of people still appear to be running the course on 8/I's, I think it would be possible to add the Programmer's Panel as a "DO-IT-YOURSELF" Kit to an 8-I, as we discussed. If I can obtain just the panel I will develop the hardware for an 8I. Perhaps it should be included as one of the optional experiments. In any case, I do need the prints for the 8-e version, since we need to describe in detail what happens (e.g., register contents at each step for each instruction). In fact, as you indicated it would be best to loan us one for the 8-f to try all these things out. As you know, describing in detail how to use a piece of hardware you haven't got is difficult, especially if you are trying to reach the un-



Mr. Alan Mowbray  
Progress Letter #9  
August 21, 1974  
Page 3

knowledgeable student.

In summary, the kit design needs to be finalized as outlined above, the I/O Access box needs to be finished and placed in production, some attention is needed on the H309 panels and the board socket problems need to be addressed.

Progress is being made in spite of not having all the materials available. I will have a student available to do the final experiment drawings this summer. Most of the drawings for the text portion on logic have been redone, and I have to rough drawings for the register-transfer logic description of the 8e/f processor. In addition, the Southern Tech situation is shaping up nicely. We need to finalize these arrangements and will be working essentially full-time this summer on the course and things should begin to move.

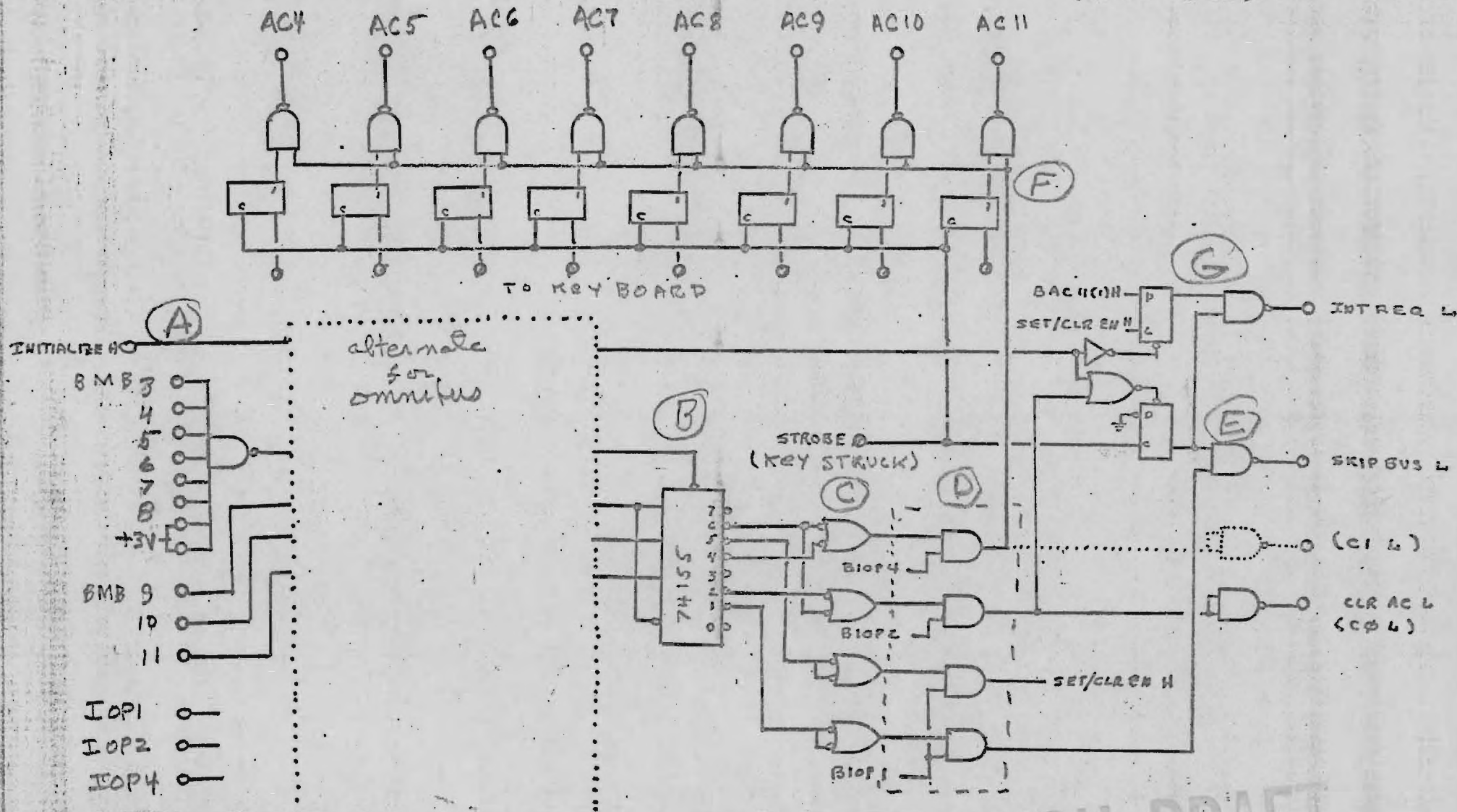
Sincerely,

Don S. Harmer  
Professor of Physics and Nuclear  
Engineering

DSH:lsg

This report is a recompilation from  
notes, telephone logs, and informal  
letters.

# KEYBOARD (E+T BUS)



ROUGH DRAFT